PCT/GB99/02288

The Patent Office

INVESTOR IN PEOPLE

The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP10 8QQ

GB99/2288

5

09/743713

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., plc, P.L.C. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.

Signed

Dated 12 AUG 1999

PATENTS ACT 1977

16JUL98 E376090-1 D02626
P01/7700 25.00 - 9815370.3

PATENTS FORM No 1/77

THE PATENT OFFICE
16 JUL 1998
RECEIVED BY POST

The Comptroller,
The Patent Office

REQUEST FOR GRANT OF A PATENT

THE GRANT OF A PATENT IS REQUESTED BY THE UNDERSIGNED ON THE BASIS OF THE PRESENT APPLICATION

I Agent's reference: TURVEY

16 JUL 1998

II Title of Invention: Content Addressable Memory

9815370.3

III Applicant:
Name: TURVEY, Douglas Philip
Country: UK
Address: Surrey, (10 261500)
34 Meadrow, Godalming GU7 3HT

IV Inventor (a) The applicant is the sole inventor

V Name of Agent: M G Harman

VI Address for Service: M G Harman & Co, Holmwood, 37 Upper Park Road, Camberley, Surrey, GU15 2EG
tel: 01276 22985

VII Declaration of Priority:

nil

VIII The Application claims an earlier date under Section 8(3), 12(6), 15(4), or 37(4):

nil

IX Check List

	A The application contains the following numbers of sheets:		B The application as filed is accompanied by:
1	Request	1 sheet	nil
2	Description	9 sheets	
3	Claims	nil AFTER DESCRIPTION	
4	Drawings	2 +2 sheets	
5	Abstract	nil	

X It is suggested that Figure No of the drawings should accompany the abstract when published

XI Signature:

Douglas Turnbull

Applicant

Content Addressed Memories

The present invention relates to memories of the type known as CAMs (Content Addressed Memories) or Associative Memories.

A conventional computer memory consists of a large number of memory locations which have sequential addresses. To access a location in such a memory, an address is supplied to the memory; the corresponding location is thereby selected, and its contents can then be accessed (ie read or written).

This works admirably when the address of the desired location (which typically stores a byte or a word) is known. However, there is a variety of situations in which data is organized in data blocks and what is wanted is a data block having particular data in a part of the block. Conventionally, the only way of finding the desired data block is to search through the data blocks one by one. A simple sequential search is the simplest procedure, but is liable to be extremely time-consuming. Various forms of directory structures or indexing can sometimes be used, but these tend to be complicated and inflexible.

To overcome these problems, a type of memory known as content addressable memory (CAM) has been proposed. This type of memory is also known as associative, because it automatically associates the desired data with the blocks containing that data.

CAMs have never achieved substantial commercial success, primarily because of the complexity of the circuitry required, compared with conventional memories. The situation is made worse because of the relatively limited range of applications for CAMs, so that CAM memories would be a relatively low-volume and high cost product even apart from the extra complexity.

The general object of the present invention is to provide an improved CAM architecture which alleviates or overcomes these problems.

According to the present invention there is provided a content addressable memory comprising a CAM control logic unit and a plurality of cells connected in a chain, each cell comprising:

a memory block coupled to a common address bus;

a comparator coupled to a common data bus and to the data interface of the memory block;

switching means coupling the data interface of the memory block with the data bus, and;

a logic block including a Match flip-flop:

the memory being operable:

in a Search phase to serially match a sequence of words on the common data bus with the contents of a sequence of addresses in the memory blocks of the cells; and

in an Access phase, to render the cells matched in the Search phase serially available for access via the common address and data buses.

A CAM embodying the invention will now be described, by way of example, with reference to the drawings, in which:

Fig. 1 is a general block diagram of the system;
Fig. 2 is a more detailed block diagram of a cell of the system; and
Fig. 3 is a detailed block diagram of the logic block of a cell of the system.

Referring to Fig. 1, the CAM consists primarily of a chain of identical cells 10, with a CAM control unit 11, which is coupled to the top of the chain of logic blocks 10.

Each cell (Fig. 2) contains a respective memory block 12, which may use say DRAM (with conventional refresh arrangements operating on each block) or SRAM. The size of each memory block may be say 64 kbytes, though obviously the number of locations may vary and the word length may also be varied (eg to 2 or 4 bytes). The number of blocks will normally be large, typically of the order of 10^5.

The cells 10 are coupled to a 1-byte data bus DATA, a 16-bit address bus ADD, and a 1-bit R/W control bus R/W. The address and data bus widths match the number of locations in the memory block in each cell and the size of each location. If the CAM forms part of a computer system, these buses may be general system-wide buses, or parts of such system buses. A system address bus

may be considerably wider than 16 bits; the bottom 16 bits will then be used as the address bus ADD for the CAM. Similarly, the R/W control bus for the CAM may form one bit of a multi-bit system control bus.

Each cell 10 also contains a memory block 12, a logic block 13, a comparator 14, and a bidirectional switch 15. The chaining of the cells 10 is through their logic bloks 13, as shown.

The CAM will normally be implemented on an integrated circuit chip. It will be realized that such a chip can easily be designed so that several such chips can be chained to increase the size of the CAM. The control unit 11 is preferably a separate unit. However, it can be included on the CAM chip if desired; in that case, several chips can be chained by designing the controller so that only that of the first chip in the chain is enabled, with those of the following chips in the chain being disabled.

The functions of the logic blocks 13 can best be understood by considering the manner in which the CAM operates. The CAM operates in 2 modes, Search and Access. The Search phase is concerned with finding the cells containing data which matches the search criteria; the Access phase is concerned with accessing those cells to extract the associated data.

For simplicity we shall assume that each cell contains a single data block of 1024 bytes stored in its memory block. If the data block is shorter than 1024 bytes, then it is simply padded out with dummy bytes. (In fact, the dummy bytes may simply be left unused, ie never read or written, in the memory block.) The data blocks also all have the same structure, which we shall take as consisting of a number of key fields Key1, Key2, Key3, etc and a number of data fields Data1, Data2, Data3, etc. Although the division of the data blocks into fields is identical for all data blocks, the number of fields, their sizes, and their locations in the data block can be chosen arbitrarily. We shall also assume that all fields are integral numbers of bytes long.

For a search, a set of key fields is defined, together with a set of search contents for those fields. Each search field is fed to the cells in turn, and for each search field, the search bytes are fed to the cells sequentially. Thus each cell receives a sequence of search bytes on the DATA bus, each accompanied on the ADD bus by its address in the data block. Further, the CAM control unit 11

feeds a Search control signal to all cells, via the chain of logic blocks 13 in the cells.

Each time a byte address is fed to a cell in the Search phase, the corresponding location in the cell's memory block 12 is read. This reads out a byte from one of the key fields in the data block stored in the cell. The stored byte is passed to the compare unit 14, which is also fed with the search byte on the data bus DATA. (Switch 15 is disabled at this time, isolating the DATA bus from the data output of the memory block.) The compare unit 14 compares the 2 bytes fed to it, and produces a Hit or Miss signal, depending on whether the search byte and the stored byte match or don't match.

The logic block 13 of each cell contains a Match flip-flop. The CAM control unit 11 initially sets the Match flip-flops of all cells to the Hit state. In each cell, each time a comparison is made by the comparator 14 between a search byte and a stored key-field byte, the result (Hit or Miss) is fed to the Match flip-flip 16. If the stored byte and the data byte don't match, the Miss output from the comparator clears this flip-flop to Miss. After the succession of search bytes in the different key fields has been run, the Match flip-flop will remain in the Hit state only if all bytes match.

If desired, a MASK bus (not shown) can be included. The MASK bus will have the same width as the DATA bus; if the system data bus is wide enough, a second byte on this bus can be used as the MASK bus. The MASK bus will be coupled to the comparators in all cells 10, and each bit on the MASK bus will determine whether or not the comparator compares the corresponding bits of the search and stored bytes. Thus bit-level rather than merely byte-level searching and matching can easily be implemented if desired. This allows tighter packing of the key fields if many of them are less than 1 byte long.

At the end of the Search phase, therefore, the Match flip-flop of each cell of the CAM will still be in the Hit state if a full match has been achieved for that cell, but will have been set to Miss if any failure of the matching has occurred for that cell.

Once the Search phase is finished, the Access phase follows. For this, the CAM control unit 11 sends an Access signal to the chain. At the start of this phase, there will be some unknown number of cells with their Match flip-flops

still set at Hit. These are coupled in a logic chain through the logic blocks 13. The topmost of these cells is enabled by the Access signal and its Match flip-flop; this enabled cell disables all the following cells in the chain.

This enabled cell can be accessed over the address, data, and read/write buses ADD, DATA, and R/W. The address bus is coupled to the memory block 12 of the cell, and the switch 15 is enabled by the logic block 13, coupling the data bus to the memory block data path. The R/W signal is, in this phase, passed through the logic block 13 to the memory block 12, to control whether reading or writing occurs. An arbitrary number of data fields can therefore be read or written, by sending the addresses of those fields to the cells in turn, and for each data field, sending the data bytes to the cells or reading them from the cells sequentially.

Once the accessing of the enabled cell is complete, the CAM control unit 11 sends a signal down the chain of cells to clear the topmost Match flip-flop to Miss. The next Match flip-flop at Hit will then enable its cell, and the the data block in that cell can then be accessed. The process continues in the same way with each cell in the chain with its Match flip-flop set being enabled in turn, until all matching data blocks have been accessed.

Obviously not all data blocks found in the Search phase need be accessed in the same way. Once accessing of a block has started, the accessing may if desired be made dependent on data read from the block.

The number of matches (if any) is in general not predictable. A return line may be provided from the end of the chain of cells back to the CAM control unit 11, and arranged to change state when all Match flip-flops in the chain have been cleared to Miss.

Alternatively, the end of the sequence can be determined by software. When the end of the sequence of matching blocks is reached, any attempt to carry on reading data blocks will result in no read occurring. The data bus will therefore return its inactive state (hex-FF if it has pull-up resistors). So to determine when all matching blocks have been read, the system monitors for this data byte. To prevent genuine data bytes with the value hex-FF from being misinterpreted as the end of the set of matching blocks, a standard byte address

is chosen in all data blocks and a non-FF-hex byte is deliberately included in that address in every data block.

Once a cell has been selected, it can of course be written as well as read. So a selected cell can simply have a complete new data block written into it, or its contents can be inspected and various parts of it changed, possibly in dependence on what it contains.

It has been assumed above that each cell contains a single data block. The optimum operating condition will be with a data block size which matches the memory block size. A data block must be confined within a single cell, so if the data blocks vary in size, the maximum size must not be larger than the cell size. However, if the data block size is considerably smaller than the cell size, it would be possible to pack 2 (or more) short data blocks into a cell. The division of the cell into such data blocks would be essentially a matter for software control. This would of course mean that a search would have to be done in 2 stages, searching first say the top halves of the cells for data block matches, and then the second halves of the cells for any further data block matches.

It is also possible for two or more data blocks to be linked together by suitable software techniques, so that a set of blocks so linked can be accessed as a group. A set of blocks so linked can be regarded as a data block whose size is not limited by the cell size.

Reviewing the system, therefore, it is evident that cells can be searched on any fields, and any cells matching the search can be read, modified, or completely rewritten. Although the operation was described above in terms of key and data filds, the division of the data block structure into such fields is purely notional; any bytes can be treated as key bytes, data bytes, or both.

It is not possible to access a cell by its position in the chain. However, this does not mean that a cell can become inaccessible. If a cell is to be effectively cleared, a standard byte address can be chosen for all data blocks, and filled with one data value if the data block in that cell is valid and another data value if the data block in the cell is cleared, ie invalid. A search for cells with the second data value in that location will then retrieve all empty data blocks in sequence. If all else fails, a search with no search fields will select all cells, which can then be written into sequentially. (That is, the Search phase

is entered to set all Match flip-flops to Hit, and the Access phase is them entered immediately.)

Referring now to Fig. 3, the logic and control circuitry of the cell 10 will be described in more detail. Each cell is, as described above, coupled to the system data bus DATA and the system address bus ADD. There is also a single line chaining all the cells together. Apart from this single line, the control signals to the cells are preferably all carried on the system control bus CONT, which is coupled to all cells. In many instances, the system CPU can conveniently be arranged to generate these signals, so that this CPU constitutes the CAM control unit 11. In addition, a further line of the control bus may be coupled to the first cell as the start of the chain line through all cells. (As discussed above, a return line from the end of the chain line may usually be dispensed with.)

Fig. 3 shows the control logic unit 13 of a typical cell 10 in more detail. A memory control unit 20 is fed with three control bus signals R/W, S/A, and EN. The R/W signal is a read/write signal, which is used to determine whether the cell memory block 12 is to be read from or written to in the access phase of the operation of the CAM. The S/A signal determines whether the CAM operates in the search or access modes. The EN signal is an enable signal, which determines whether the CAM is enabled or not. When enabled, it is assumed that the other devices coupled to the system buses are disabled; when the CAM is disabled, other devices coupled to the system buses may be enabled without their operation being affected by the existence of the CAM.

It will be realized, of course, that the CAM (or the integrated circuit chips carrying the CAM) may be assigned a control address, so that the CAM can be enabled and disabled by means of such a control address. In effect, this means that the CAM is enabled by a suitable signal combination (the control address) on a plurality of control bus lines, but that set of control bus lines will be shared with many other devices in the system, rather than the CAM requiring a dedicated control bus line.

The memory logic block 20 generates three output signals. There is a control signal to the bidirectional switch 15, which is used to couple the memory block 12 and the data bus DATA to the comparator 14 in the search phase and the memory block to the data bus in the access phase. There is a R/W signal to

the memory block 12 to set it for reading in the search phase and for reading or writing, as the case may be, in he access phase. And there is an Enable signal to the memory block 12, to enable it in the search phase and, if the cell is the selected cell, in the access phase.

The cell also includes a chain line coupling switch 21 which couples the chain line into the cell, C-IN, to the chain line out of the cell, C-OUT. This switch in effect either couples the chain in line C-IN to the chain out line C-OUT or imposes a logic 0 on the chain out line C-OUT. It can conveniently be implemented as a pair of bidirectional switches, one connected between the chain in and chain out lines and the other connecting the chain out line to earth (logic 0), with a control signal driving one switch directly and the other via an inverter.

The control logic unit 13 also includes the match flip-flop 16, which has already been discussed. This flip-flop has an input from the comparator 14, as described above. It is also fed with a further signal RST from the system control bus CONT. This signal RST is a reset signal which is used to reset all match flip-flops in the CAM to the match state at the beginning of the search phase.

Finally, there is a Next control block 22. This is fed from the match flip-flop 16 and from the chain in line C-IN, and controls the chain line switch 21. If the signal on the chain in line is a logic 1 (which indicates that all cells above the present cell in the chain have been dealt with), and if the match flip-flop remains set at the end of the search phase, then the present cell is the next one to be accessed. The Next control block 22 enables the memory control block 20, and sets the switch 21 to logic 0. If the match flip-flop has been cleared, however, then the Next control block automatically disables the memory control block 20 (so that the cell cannot be accessed) and sets the switch 21 to pass the signal on the chain in line on to the chain out line.

The Next control block is also fed with the fifth of the five signals from the system control bus, NXT, and further includes a flip-flop. If the cell is the currently selected one (ie the match flip-flop is still at match and all cells above it in the chain have been accessed), then the NXT signal effectively disables the present cell. The Next control block 21 sets the switch 21 to pass the logic 1 on to the next cell down the chain, and disables the memory control logic 20 so

that the cell cannot be accessed again. The flip-flop of the Next control block is used to capture the state of the incoming C-IN signal and prevent the NXT signal from propagating down the chain. The Next control block therefore also has an output to the match flip-flop 16.

Claims

Any novel and inventive feature or combination of features specifically disclosed herein within the meaning of Article 4H of the International Convention (Paris Convention).

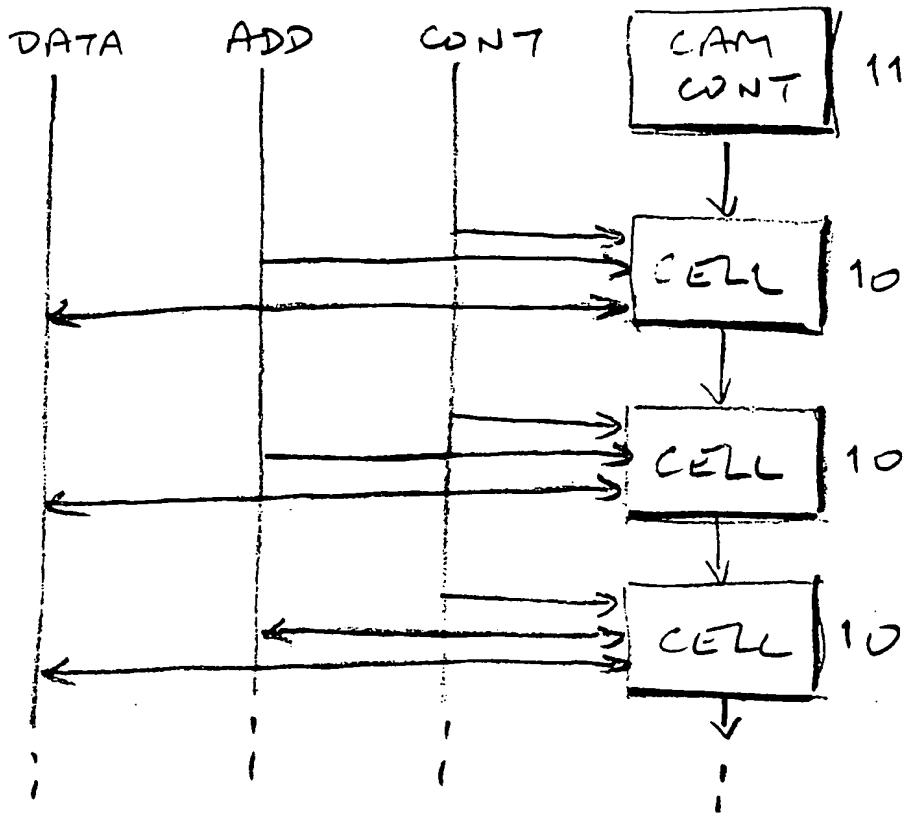

DATA
ADD
CONT
CAM CONT
11
CELL
10
CELL
10
CELL
10

FIG. 1

FIG. 2

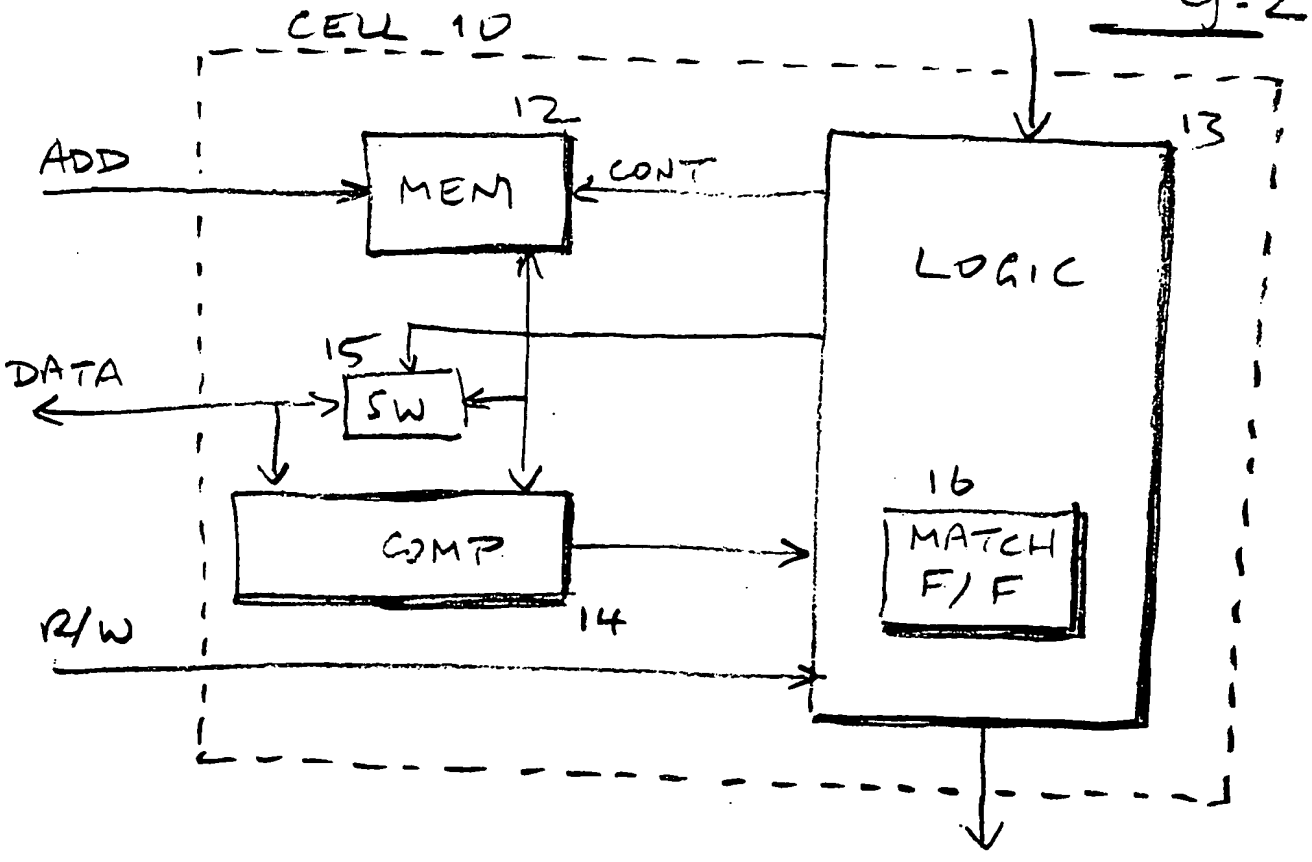
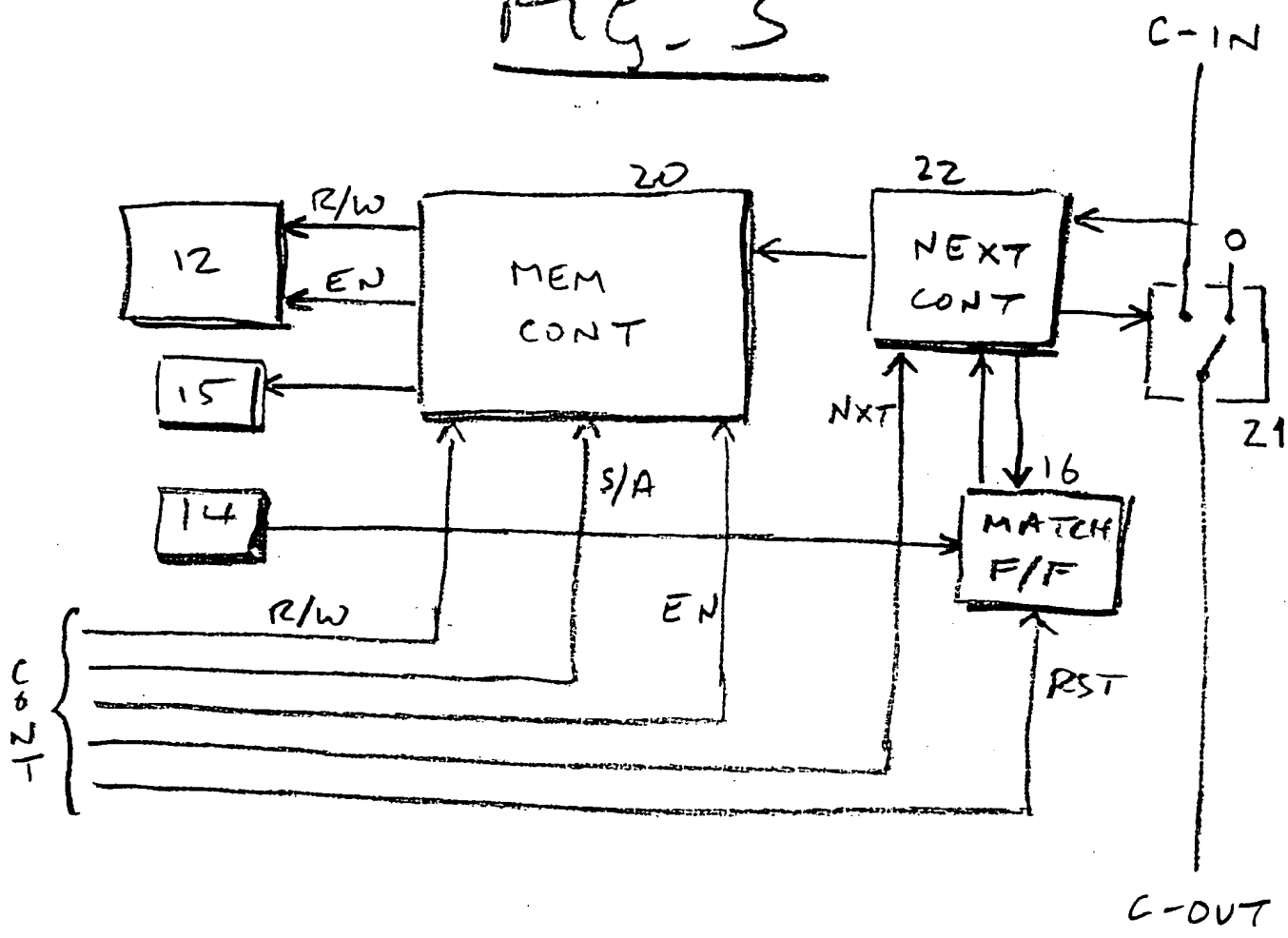

CELL 10
12
13
ADD
MEM
CONT
LOGIC
15
DATA
SW
16
COMP
MATCH F/F
14
R/W

FIG. 3


C-IN
20
22
R/W
12
EN
MEM CONT
NEXT CONT
0
15
NXT
16
Z1
S/A
14
MATCH F/F
R/W
EN
CONT
RST
C-OUT